

CLAIM AMENDMENTS

1 Claims 1-35 (Cancelled).

1 36. (New) A computer-implemented method for testing an operating system, comprising:  
2 (a) causing said operating system to create a process having at least a first thread and a  
3 second thread;  
4 (b) causing two or more test functions to be executed in said first thread, wherein each  
5 test function of said two or more test functions is a different test function;  
6 (c) causing said two or more test functions to be executed in said second thread; and  
7 (d) repeating steps (a), (b), and (c) to cause said operating system to create one or  
8 more additional processes each having at least two threads and to cause said two or  
9 more test functions to be executed in each of said at least two threads.

1 37. (New) A method as recited in Claim 36, wherein said two or more test functions are  
2 executed in parallel in each of said first thread, said second thread, and said at least two  
3 threads.

1 38. (New) A method as recited in Claim 36, wherein said two or more test functions are  
2 executed sequentially in each of said first thread, said second thread, and said at least  
3 two threads.

1 39. (New) A method as recited in Claim 36, wherein each test function of said two or more  
2 test functions is executed substantially simultaneously by each of said first thread, said  
3 second thread, and said at least two threads.

1 40. (New) A method as recited in Claim 36, wherein each test function of said two or more  
2 test functions is executed entirely in a particular thread before said each test function  
3 begins execution in another thread.

1 41. (New) A method as recited in Claim 36, wherein each test function of said two or more  
2 test functions is executed randomly in each of said first thread, said second thread, and  
3 said at least two threads.

1 42. (New) A method as recited in Claim 36, further comprising the computer-implemented  
2 step of:  
3 repeating step (d) until a total number of processes created by said operating system is  
4 at least equal to a predetermined number of processes.

1 43. (New) A method as recited in Claim 36, further comprising the computer-implemented  
2 step of:  
3 executing step (d) such that a total number of threads per process is at least equal to a  
4 predetermined number of threads per process.

1 44. (New) A method as recited in Claim 36, further comprising the computer-implemented  
2 step of:  
3 executing step (d) such that a total number of times that each test function of said two  
4 or more test functions is executed by each thread is at least equal to a  
5 predetermined number of times.

1 45. (New) A method as recited in Claim 36, further comprising the computer-implemented  
2 step of:  
3 repeating step (d) until a plurality of resources available to said operating system are  
4 exhausted.

1 46. (New) A computer-readable medium for testing an operating system, the computer-readable  
2 medium carrying instructions which, when executed by one or more processors, cause  
3 performance of the steps of:  
4 (a) causing said operating system to create a process having at least a first thread and a  
5 second thread;

- 6 (b) causing two or more test functions to be executed in said first thread, wherein each  
7 test function of said two or more test functions is a different test function;  
8 (c) causing said two or more test functions to be executed in said second thread; and  
9 (d) repeating steps (a), (b), and (c) to cause said operating system to create one or  
10 more additional processes each having at least two threads and to cause said two or  
11 more test functions to be executed in each of said at least two threads.

1 47. (New) A computer -readable medium as recited in Claim 46, wherein said two or more  
2 test functions are executed in parallel in each of said first thread, said second thread,  
3 and said at least two threads.

1 48. (New) A computer -readable medium as recited in Claim 46, wherein said two or more  
2 test functions are executed sequentially in each of said first thread, said second thread,  
3 and said at least two threads.

1 49. (New) A computer -readable medium as recited in Claim 46, wherein each test  
2 function of said two or more test functions is executed substantially simultaneously by  
3 each of said first thread, said second thread, and said at least two threads.

1 50. (New) A computer -readable medium as recited in Claim 46, wherein each test  
2 function of said two or more test functions is executed entirely in a particular thread  
3 before said each test function begins execution in another thread.

1 51. (New) A computer -readable medium as recited in Claim 46, wherein each test  
2 function of said two or more test functions is executed randomly in each of said first  
3 thread, said second thread, and said at least two threads.

1 52. (New) A computer -readable medium as recited in Claim 46, further comprising  
2 instructions which, when executed by the one or more processors, cause performance of  
3 the step of:  
4 repeating step (d) until a total number of processes created by said operating system is  
5 at least equal to a predetermined number of processes.

1 53. (New) A computer -readable medium as recited in Claim 46, further comprising  
2 instructions which, when executed by the one or more processors, cause performance of  
3 the step of:  
4 executing step (d) such that a total number of threads per process is at least equal to a  
5 predetermined number of threads per process.

1 54. (New) A computer -readable medium as recited in Claim 46, further comprising  
2 instructions which, when executed by the one or more processors, cause performance of  
3 the step of:  
4 executing step (d) such that a total number of times that each test function of said two  
5 or more test functions is executed by each thread is at least equal to a  
6 predetermined number of times.

1 55. (New) A computer -readable medium as recited in Claim 46, further comprising  
2 instructions which, when executed by the one or more processors, cause performance of  
3 the step of:  
4 repeating step (d) until a plurality of resources available to said operating system are  
5 exhausted.

1 56. (New) A mechanism for testing an operating system, the mechanism comprising a  
2 memory storing instructions which, when executed by one or more processors, cause  
3 performance of the steps of:  
4 (a) causing said operating system to create a process having at least a first thread and a  
5 second thread;  
6 (b) causing two or more test functions to be executed in said first thread, wherein each  
7 test function of said two or more test functions is a different test function;  
8 (c) causing said two or more test functions to be executed in said second thread; and  
9 (d) repeating steps (a), (b), and (c) to cause said operating system to create one or  
10 more additional processes each having at least two threads and to cause said two or  
11 more test functions to be executed in each of said at least two threads.

1 57. (New) A mechanism as recited in Claim 56, wherein said two or more test functions  
2 are executed in parallel in each of said first thread, said second thread, and said at least  
3 two threads.

1 58. (New) A mechanism as recited in Claim 56, wherein said two or more test functions  
2 are executed sequentially in each of said first thread, said second thread, and said at  
3 least two threads.

1 59. (New) A mechanism as recited in Claim 56, wherein each test function of said two or  
2 more test functions is executed substantially simultaneously by each of said first thread,  
3 said second thread, and said at least two threads.

1 60. (New) A mechanism as recited in Claim 56, wherein each test function of said two or  
2 more test functions is executed entirely in a particular thread before said each test  
3 function begins execution in another thread.

1 61. (New) A mechanism as recited in Claim 56, wherein each test function of said two or  
2 more test functions is executed randomly in each of said first thread, said second thread,  
3 and said at least two threads.

1 62. (New) A mechanism as recited in Claim 56, wherein the memory further comprises  
2 one or more instructions which, when executed by the one or more processors, cause  
3 performance of the step of:  
4 repeating step (d) until a total number of processes created by said operating system is  
5 at least equal to a predetermined number of processes.

1 63. (New) A mechanism as recited in Claim 56, wherein the memory further comprises  
2 one or more instructions which, when executed by the one or more processors, cause  
3 performance of the step of:  
4 executing step (d) such that a total number of threads per process is at least equal to a  
5 predetermined number of threads per process.

1    64.    (New) A mechanism as recited in Claim 56, wherein the memory further comprises  
2            one or more instructions which, when executed by the one or more processors, cause  
3            performance of the step of:  
4            executing step (d) such that a total number of times that each test function of said two  
5                    or more test functions is executed by each thread is at least equal to a  
6            predetermined number of times.

1    65.    (New) A mechanism as recited in Claim 56, wherein the memory further comprises  
2            one or more instructions which, when executed by the one or more processors, cause  
3            performance of the step of:  
4            repeating step (d) until a plurality of resources available to said operating system are  
5                    exhausted.